Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 00 (2016) 000–000

Procedia Engineering

www.elsevier.com/locate/procedia

25th International Meshing Roundtable (IMR25)

A variational framework for high-order mesh generation

Michael Turner[a], Joaquim Peiró[a], David Moxey[a,*]

[a]*Department of Aeronautics, South Kensington Campus, Imperial College London, London SW7 2AZ, U.K.*

Abstract

The generation of sufficiently high quality unstructured high-order meshes remains a significant obstacle in the adoption of high-order methods. However, there is little consensus on which approach is the most robust, fastest and produces the 'best' meshes. We aim to provide a route to investigate this question, by examining popular high-order mesh generation methods in the context of an efficient variational framework for the generation of curvilinear meshes. By considering previous works in a variational form, we are able to compare their characteristics and study their robustness. Alongside a description of the theory and practical implementation details, including an efficient multi-threading parallelisation strategy, we demonstrate the effectiveness of the framework, showing how it can be used for both mesh quality optimisation and untangling of invalid meshes.

Keywords: high-order mesh generation; variational mesh generation; energy functional; numerical optimization.

1. Introduction

The high accuracy and low dispersion and diffusion errors of high-order methods makes them ideal candidates for the unsteady simulations in areas such as aeroacoustics, turbulent flow and combustion. However, it is widely accepted that the lack of robust high-order mesh generators for complex geometries is still a major bottleneck in the wider adoption of these methods within industrial practice [1,2]. The generation of a curvilinear unstructured mesh is achieved in general through the transformation of a coarse straight-sided linear mesh, which can be obtained by any of the well-established unstructured mesh generators, onto a boundary conforming high-order mesh. The challenge in this approach is accommodating boundary curvature into the mesh interior so that the resulting curvilinear elements are valid; that is, they do not self-intersect. A secondary problem is generating meshes that are of sufficient quality to allow simulations that retain the accuracy and convergence properties of the underlying high-order discretisation.

As high-order methods have gained interest within the community, there is an increasing number of approaches used to achieve this linear-to-curvilinear transformation. They can be broadly classified into two groups: optimisation of a functional in which distortion of elements or a mesh energy is minimised, and solid body formulations where the mesh is modelled as an elastic body and an elliptic PDE is then solved to obtain a displacement within the domain. In the first category, Dey et al. [3] investigated the use of the scaled Jacobian as a distortion metric to drive local

[*] Corresponding author.
E-mail address: d.moxey@imperial.ac.uk

curvature-based refinement and mesh optimisation. Sherwin and Peiro [4] use a spring analogy to position points in curves and surfaces to minimize a deformation energy. Toulorge et al [5] investigate the use of a logarithmic barrier technique in combination with an unconstrained optimisation of the Jacobian and theoretical bounds obtained through Bézier functions. A number of works by Roca and collaborators have investigated the use of shape distortion optimisation to produce unstructured curvilinear meshes, e.g. [6–8]. In the category of solid body deformation, there have been a similar number of investigations: a linear elasticity approach is used by Xie et al. [9], a non-linear elasticity by Persson and Peraire [10], and a Winslow formulation by Fortunato and Persson [11]. Additionally, some of the authors have also investigated a thermo-elastic model [12] in order to control the quality generated by elastic models.

The purpose of this work is to demonstrate that many of these approaches can be reformulated in a generalised framework, which is based on a variational approach to curvilinear mesh generation. In a variational setting, a functional defining a measure of energy over the mesh, and which takes as its arguments the mesh displacement and its derivatives, is minimised using a nonlinear optimisation strategy. We will show how this framework can be used for the purposes of both untangling invalid curvilinear meshes and for mesh optimisation of existing valid meshes.

The motivation of this study comes in the development of similar approaches in the linear mesh generation community, for example in references [13,14]. These have been under development as early as the 1970s, where Felippa investigated the applicability of direct energy searches to the problem of mesh generation [15]. However, these methods are yet to be fully investigated in the context of high-order mesh generation and optimisation, although they have been briefly mentioned in previous work [16]. In this variational framework, we can not only consider and compare many of the existing approaches to high-order mesh generation, but capitalise on a number of mathematical and technical advantages that this framework offers. Firstly, we may examine a number of different functionals and compare their behaviour in terms of speed of convergence and the resulting mesh quality obtained. From the standpoint of analysis, the use of an energy functional guarantees the existence of a minimum under certain conditions of the behaviour of the functional such as convexity or polyconvexity, as noted by Evans [17]. Although we do not study this property in detail here (see, for example, Huang and Russell [14] and Garanzha [18]), this approach adds robustness to the method through these theoretical guarantees. From a technical perspective, the implementation of a nonlinear optimisation strategy for these functionals is arguably easier than, for example, a finite element discretisation of linear or nonlinear PDEs. We will show how this variational formulation permits highly efficient and effective multi-threading parallel execution, thus allowing for the optimisation of large high-order meshes in minutes on a modern desktop workstation.

The paper is structured as follows. In section 2, we give a brief formulation of the problem in terms of the underlying solid mechanics formulation and outline the four energy functionals that we will investigate in this work, which overlap with a large number of studies based around high-order mesh generation. In section 3 we describe details of the practical implementation needed in this variational setting such as discretisation and non-linear optimisation, with a brief discussion on untangling in section 4. Section 5 then examines the application of this method to a number of two- and three-dimensional problems, examining the meshes obtained by each method, the number of iterations and computational time needed for convergence. We finalise the paper in section 6 with a brief overview and outlook to future work and improvements.

2. Background and formulation

The premise of this work is to examine the generation of a boundary-conforming high-order mesh that is obtained through a deformation. The starting point is a straight-sided high-order mesh $\Omega = \bigcup_{e=1}^{N_{\mathrm{el}}} \Omega^e$ composed of N_{el} conformal elements, which is equipped with geometry-conforming displacements at the boundary of the domain. We view Ω as a solid body, so that the displacements at the boundary induce a deformation in the interior of the domain. The end result should be a valid high-order mesh Ω_*, where the interior elements have been deformed to accommodate the displacement at the boundary and improve the quality of the resulting elements.

From a solid mechanics perspective, we may construct a model that, for example, represents linear or nonlinear elasticity. This often appears in the form of an elliptic partial differential equation which may be solved to calculate the displacement $\boldsymbol{u}$ between the original and deformed states. As mentioned in the introduction, these approaches have both previously been examined in the context of high-order mesh generation. However the purpose of this work

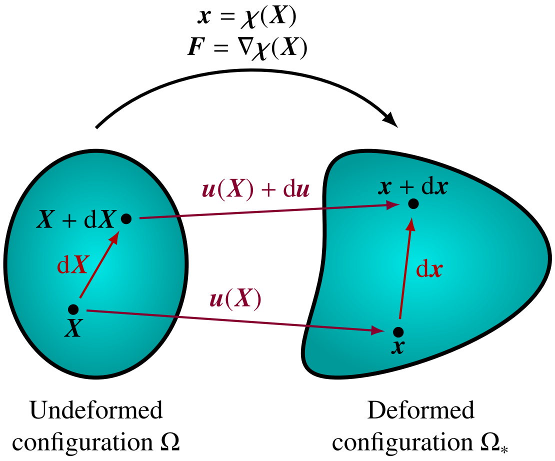


Fig. 1: Notation used for the deformation mapping $\chi : \Omega \to \Omega_*$.

is to instead consider a *variational* approach, in which we aim to find a displacement $\boldsymbol{u}$ that minimizes an energy functional

$$\mathcal{E}(\boldsymbol{u}) = \int_{\Omega} W(\boldsymbol{u}, D\boldsymbol{u})\,\mathrm{d}\boldsymbol{X} \tag{1}$$

where $\boldsymbol{u} = \boldsymbol{x} - \boldsymbol{X}$ denotes the displacement of a point $\boldsymbol{X} \in \Omega$ from its initial straight-sided configuration to the deformed position $\boldsymbol{x} \in \Omega_*$, and $D\boldsymbol{u}$ represents its first derivatives. The problem is closed by setting a boundary condition of zero displacement at the boundary once the deformation has been imposed.

In this setting, the problem is recast into one of selecting an appropriate (and generally non-linear) energy function W and a numerical method for the arising nonlinear optimisation problem. In this section, we first describe the solid mechanics preliminaries that underpin this formulation and present a number of choices for W which encompass previous methods for high-order mesh generation.

2.1. *Solid mechanics preliminaries*

To introduce the solid mechanics preliminaries that are used to define the variational problem, we follow a similar notation to that of reference [19]. The deformation of an initial, or undeformed, configuration of a solid, denoted by Ω, with boundary $\partial\Omega$ into a new, or deformed, configuration, Ω_*, with boundary $\partial\Omega_*$ is described mathematically by a mapping $\chi : \Omega \to \Omega_*$. Coordinates $\boldsymbol{X}$ in the initial configuration move into a point of coordinates $\boldsymbol{x}$ in the deformed configuration under the action of a mapping $\chi : \Omega \to \Omega_*$ so that $\boldsymbol{x} = \chi(\boldsymbol{X})$. The notation used to describe the mapping representing the deformation is depicted in fig. 1. The deformation gradient tensor is defined as

$$\boldsymbol{F} = \frac{\partial\chi}{\partial\boldsymbol{X}} = \nabla\chi,$$

which relates elemental vectors and differentials in the configurations according to $\mathrm{d}\boldsymbol{x} = \boldsymbol{F}\,\mathrm{d}\boldsymbol{X}$. The determinant $J = \det \boldsymbol{F}$, is referred to as the Jacobian of the mapping, and represents volumetric deformations so that $dv = J\,dV$, where dv and dV denote elemental volumes in the deformed and undeformed configurations respectively. General measures of deformation are represented by the strain tensors. Let us consider two elemental vectors, $\mathrm{d}\boldsymbol{X}_1$ and $\mathrm{d}\boldsymbol{X}_2$, in the undeformed configuration that deform into the elemental vectors $\mathrm{d}\boldsymbol{x}_1$ and $\mathrm{d}\boldsymbol{x}_2$, respectively. The right Cauchy-Green tensor, $\boldsymbol{C}$, characterizes their scalar product $\mathrm{d}\boldsymbol{x}_1 \cdot \mathrm{d}\boldsymbol{x}_2 = \mathrm{d}\boldsymbol{X}_1 \cdot \boldsymbol{C}\,\mathrm{d}\boldsymbol{X}_2$, which can be written in terms of the deformation gradient as $\boldsymbol{C} = \boldsymbol{F}^{\top}\boldsymbol{F}$.

2.2. Forms of the energy functional

In this section, we outline a key contribution of this work, by highlighting how many of the existing curvilinear mesh generation methods can be unified in a variational setting through the definition of an energy functional.

2.2.1. Linear elasticity energy

A number of works [9,12,20] have examined the use of a linear elastic analogy in the context of high-order mesh generation. This takes the form of an elliptic PDE: $\nabla \cdot (\boldsymbol{\lambda}\,\mathrm{tr}(\boldsymbol{E})\boldsymbol{I} + \mu\boldsymbol{E}) = -\boldsymbol{f}$, where $\boldsymbol{E} = \frac{1}{2}(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^\top)$, λ and μ are the Lamé constants, we assume that there are no body forces so that $\boldsymbol{f} = \boldsymbol{0}$ and a displacement is prescribed at the boundary as a Dirichlet condition to close the problem. However, we may alternatively view this as the Euler-Lagrange equation of the functional (1) with W given by

$$W = \frac{1}{2}\lambda\,[\mathrm{tr}(\boldsymbol{E})]^2 + \mu\,\boldsymbol{E} : \boldsymbol{E},$$

where the double product or Frobenius product of two tensors is defined as $\boldsymbol{A} : \boldsymbol{B} = \mathrm{tr}(\boldsymbol{A}^\top\boldsymbol{B})$. However, we note that this form leads to the linear elasticity formulation for small deformations, but it does not satisfy the growth condition that $W \to \infty$ when $J \to 0^+$, which is required to prevent the inversion of the mesh [21]. By defining $\kappa > 0$ as the bulk modulus, a modified version of this energy that performs better for large compressive strains, i.e. when $J \to 0^+$, is

$$W = \frac{\kappa}{2}(\ln J)^2 + \mu\,\boldsymbol{E} : \boldsymbol{E}.$$

2.2.2. Isotropic hyperelasticity energy

We now consider a nonlinear hyperelastic formulation that aligns with the work of [10]. We note that if the material is isotropic, i.e. the constitutive behaviour is identical in any direction, then the energy must be a function of the invariants of the right Cauchy-Green tensor, $\boldsymbol{C}$, only. This is written as $W(\boldsymbol{C}(\boldsymbol{X}), \boldsymbol{X}) = W(I_1^{\boldsymbol{C}}, I_2^{\boldsymbol{C}}, I_3^{\boldsymbol{C}}, \boldsymbol{X})$ where the invariants of $\boldsymbol{C}$ are

$$I_1^{\boldsymbol{C}} = \mathrm{tr}(\boldsymbol{C})\,; \quad I_2^{\boldsymbol{C}} = \mathrm{tr}(\boldsymbol{C}\boldsymbol{C}) = \mathrm{tr}(\boldsymbol{C}^\top\boldsymbol{C}) = \boldsymbol{C} : \boldsymbol{C}\,; \quad I_3^{\boldsymbol{C}} = \det(\boldsymbol{C}) = J^2$$

A simple case of isotropic hyperelastic material is the compressible neo-Hookean material, as considered in [10], and its strain energy is given by

$$W = \frac{\mu}{2}(I_1^{\boldsymbol{C}} - 3) - \mu\ln J + \frac{\lambda}{2}(\ln J)^2$$

where λ and μ are the material constants.

2.2.3. Winslow equation energy

The Winslow equations are second-order non-linear elliptic partial differential equations which are obtained by enforcing the computational coordinates to be harmonic. These have long been used in the smoothing of linear meshes and have recently been used in the application of optimisation and untangling of high-order meshes [11]. They can be recast into a variational format by again viewing them as the Euler-Lagrange equation of the functional (1) with

$$W = \frac{||\boldsymbol{F}||_f^2}{J} \tag{2}$$

as shown, for example, in [22]. Here $||\cdot||_f = \sqrt{\mathrm{tr}(\boldsymbol{F}^\top\boldsymbol{F})}$ denotes the Frobenius norm.

2.2.4. Energy as a measure of distortion

The final functional we consider here is a shape distortion measure that has been used in both linear [23] and curvilinear [6–8] mesh generation. We define (1) using

$$W = \frac{||\boldsymbol{F}||_f^2}{d|J|^{2/d}} \tag{3}$$

where d is the dimension of the mesh (in this work, $d = 2$ or 3). An interesting point, which to the best of our knowledge has not been noted elsewhere in the literature, is the similarity between this distortion measure and the Winslow functional. Whilst the denominator of eq. (3) ensures a different result for 3D meshes, we note that in the presence of a positive Jacobian in 2D, eqns. (2) and (3) are *equivalent* for the purpose of optimisation.

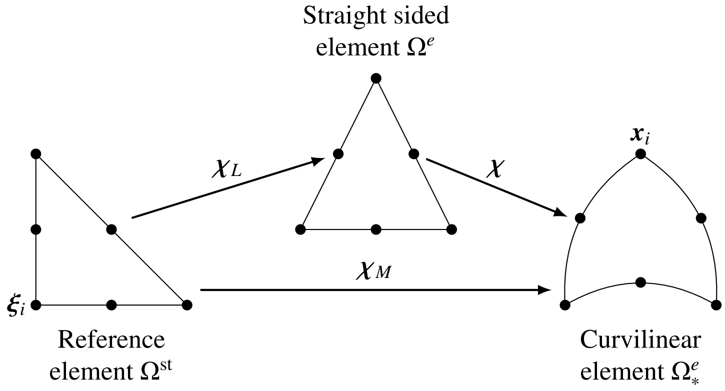


Fig. 2: The relation between the reference, straight sided and curvilinear elements and the respective mappings between them.

3. Implementation of the framework

After outlining the variational framework and the energy functionals to be investigated, we consider the practical numerical implementation in the optimisation of the functional (1). For large meshes comprising millions of moving nodes, we require both an efficient nonlinear optimisation method, alongside a robust calculation of the elemental contributions to the functional. We outline these details in this section, alongside a simple parallelisation strategy that can be used to mitigate the overall computational cost on many-core machines.

We note that the implementation described here is part of a new high-order meshing tool, *NekMesh* [24], which is contained in the open-source *Nektar++* spectral/*hp* element framework [25].

3.1. High-order discretisation

The functional $\mathcal{E}(\boldsymbol{u})$ in eq. (1) defined across the domain Ω can be considered as a sum of elemental contributions

$$\mathcal{E}(\boldsymbol{u}) = \sum_{e=1}^{N_{\text{el}}} \int_{\Omega^e} W(\boldsymbol{u}(\boldsymbol{X}), \boldsymbol{D}\boldsymbol{u}(\boldsymbol{X}))\,\mathrm{d}\boldsymbol{X} = \sum_{e=1}^{N_{\text{el}}} \int_{\Omega^{\text{st}}} W(\boldsymbol{u}(\boldsymbol{X}), \boldsymbol{D}\boldsymbol{u}(\boldsymbol{X}))\,|\det(\chi_L(\boldsymbol{X}))|\,\mathrm{d}\boldsymbol{\xi}.$$

where, as shown in fig. 2, Ω^e is the straight-sided high-order element with a corresponding curvilinear element Ω^e_*. Fig. 2 also illustrates our use of a reference element Ω^{st}, which we use in customary fashion to define the spectral element basis. The nodes of the mesh, which we denote by $\boldsymbol{x}_i$, lead to a discretisation based on the standard nodal spectral element of degree p as defined in e.g. [26], on which we perform integration through the use of quadrature weights w_i with associated points $\boldsymbol{\xi}_i \in \Omega_{\text{st}}$ and differentiation through the derivative matrix $\mathcal{D}_j$ for each coordinate direction j. In this setting, the deformation tensor is then given by $\boldsymbol{F} = \nabla\chi = (\nabla\chi_L)^{-1}\nabla\chi_M$, where χ_L is the mapping between the reference element and the linear element that, for the triangles and tetrahedra we consider in this article, is a constant matrix that can be calculated as a preprocessing step. χ_M is the mapping between the reference and high-order curved element which must be continually updated as the optimisation procedure progresses. The optimisation problem then becomes one of finding $\boldsymbol{u}$, which is a function of the mesh nodes $\boldsymbol{x}_i$ that minimises the functional (1).

In this setting, the variational optimisation problem can be examined as a local one, whereby we optimise the functional by considering each node independently. We note that each mesh node is connected to a subset of elements that is typically very small in comparison to the total number of elements, which will both reduce the computational cost and allow us to parallelise the problem as we will discuss later. For example, fig. 3 shows that if we adjust the position of nodes lying on the interior quadrilateral edges, we only need to consider the evaluation of a functional that is connected to either two or four elements as denoted by the arrows, and any interior nodes only need to be considered on a single element. The minimisation of the global functional can then be considered as that of a non-linearly related set of energies, each of which belongs to a node in a mesh. Therefore we aim to minimise

$$\mathcal{E}_i(\boldsymbol{u}) = \sum_{e\subset i} \int_{\Omega^e} W(\boldsymbol{u}(\boldsymbol{y}), \boldsymbol{D}\boldsymbol{u}(\boldsymbol{y}))\,\mathrm{d}\boldsymbol{y} \tag{4}$$

where i is any non-boundary node in the mesh and $e \subset i$ denotes the subset of elements influenced by a change in the position of node i. We note that the evaluation of $\boldsymbol{F}$, which requires the Jacobian matrix $\nabla\chi_M$, can benefit from this amalgamation of elements. Each component of $\boldsymbol{F}$ is a derivative of the form $\partial_{\xi_j}\chi_i$ that is numerically calculated by the matrix-vector multiplication $\mathcal{D}_j\boldsymbol{x}_i^e$, where $\boldsymbol{x}_i^e$ are the nodes contained within each element Ω^e. Since in this process we calculate derivatives across multiple elements concurrently, we can instead view this as a matrix-matrix multiplication. We may therefore utilise an optimised BLAS call such as `dgemm` for larger polynomial orders in order to gain computational efficiency. In the presence of lower order elements where matrix sizes are far smaller, optimised libraries such as `libxsmm` [27] may yield a significant performance improvement.

3.2. Integration rule

One of the key steps in the evaluation of elemental contributions is the quadrature rule used to approximate the integral of the functional across each element. Our initial efforts used high-order nodes conforming to an α-optimised distribution on 2D and 3D triangles and tetrahedra [26]. Whilst this is a commonly used distribution and has been used in the optimisation of the shape distortion functional in [7], our experience has been that this distribution is prone to introducing instability as part of the nonlinear optimisation, since at most polynomial orders, these distributions have negative quadrature weights at the vertices of the element. We determined that these issues occur when a mesh node is moved to a point that results in a very high elemental deformation. This yields an abnormally large value of the functional at an element vertex which, when multiplied by a negative weight, results in a large negative value and causes the optimiser to locate a new minimum where one may not exist. One potential solution is the use of a larger number of quadrature points to evaluate the gradient of the deformation tensor, functional and quadrature, as is performed in e.g. [7]. Although the quadrature weights are still negative at the vertices of the triangle, there is a greater clustering of nodes in these areas which means that the large gradient of the deformation can be accurately resolved, thereby preserving positivity of the integral. Our testing showed that whilst this yielded additional stability it did come at a greatly increased computational cost.

To overcome both the cost and stability issues we propose using an alternative set of quadrature points while using high-order nodes which are distributed in the α-optimised form. We use quadrature rules proposed by Witherden and Vincent [28], which are symmetric, interior to the standard element, but crucially have positive quadrature weights for all elements at all orders. These point distribution sets also have lower numbers of points which can achieve the same level of accuracy in integration. Therefore we no longer need to over-integrate to such a large extent, resulting in a lower computational cost whilst preserving the robustness of the method. We note that there are many such quadrature rules in the literature which may also yield similar properties.

3.3. Numerical optimisation

To minimise the energy $\mathcal{E}_i$ associated with each mesh node i, a gradient decent algorithm with a truncated Newton step is used. Algorithmically, to optimise the location of a mesh node, we first evaluate the gradient vector $\boldsymbol{G}$ and Hessian matrix $\boldsymbol{H}$, which denote the derivatives of the energy with respect to the coordinate directions of the node. These are most easily obtained by moving the node i some small distance, recalculating the functional and using a finite difference approximation. The numerical approximation of the derivatives is therefore one of the most expensive parts of the optimisation procedure. We note that whilst other work [29] shows that analytic forms of these derivatives are possible, we do not consider this here and leave it as a point for future work, since the level of performance we are able to achieve within the current optimisation framework does not lead us to immediately need to find a more efficient alternative. Additionally, the use of approximate derivatives allows us to maintain a more modular framework that depends only on the specification of the function W. At step k of the optimisation, we calculate $\boldsymbol{G}$ and $\boldsymbol{H}$ and then update the spatial coordinate of the node, denoted by $\boldsymbol{x}_i$, so that

$$\boldsymbol{x}_i^{k+1} = \boldsymbol{x}_i^k - \alpha\boldsymbol{H}^{-1}\boldsymbol{G} \tag{5}$$

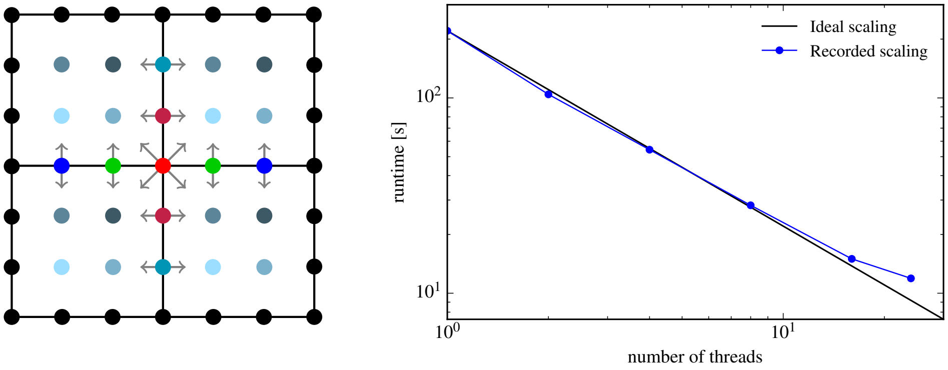


Fig. 3: *Left:* colouring scheme of 10 colours used to partition the mesh into sub-regions for parallel computing. Black vertices denote fixed (boundary) vertices that are not optimised and arrows show the elements that each vertex is connected to. Vertices that share the same colour can be optimised in parallel. *Right:* parallel scalability of sphere test case consisting of 10,615 tetrahedra with 128,254 free nodes at $P = 4$.

where $0 < \alpha \leq 1$ is a step size limiting parameter widely used to stabilise the optimisation of systems with oscillating or strong gradients, calculated via a reverse line search as described in [30]. Each node is moved one step using this procedure and then the whole process repeated over the entire set of non-boundary nodes. This procedure of optimising each node in the mesh is continued until a converge criteria is met. We set a strong convergence criterion of $\|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i^k\|_\infty / L < \varepsilon$ for all nodes i, so that the mesh is optimised once no node has moved more than a small fraction, ε, of a characteristic length, L, of the problem. We use $\varepsilon = 10^{-6}$ as a guideline for convergence.

3.4. Parallelisation

To address the issue of high associated cost of the problem, the optimisation has been implemented in parallel by formulating the global optimisation problem as a sequence of local ones as highlighted in eq. 4. Using a simple node colouring scheme, we can sort all non-boundary nodes in the mesh into subsets which are disjoint from any other subsets, allowing each optimisation to be run independently. Construction of the colouring is straightforward. We first identify a node that is currently uncoloured and assign it a new colour. We then identify all elements that are connected to this node and colour them with this same colour, and remove all nodes in these elements to avoid conflicting colours. This process is then repeated until either the mesh is coloured or we cannot colour any additional elements. At this point, we have generated a group of subsets that can be processed concurrently, for which we use the standard pthread model, together with a simple thread manager. We then re-add the removed nodes and repeat this process until all nodes are coloured. We describe an example of this in the following section.

4. Mesh untangling

Each of the functionals used to describe the energy of deformation behave asymptotically around $J = 0$, so that $W \to \infty$ as $J \to 0^+$. This is a desirable and expected property which prevents the mesh from inverting. The solid mechanics analogy would be that this behaviour prevents the interpenetration of matter. However, all have various undesirable properties when $J < 0$. Therefore they are unsuitable for any problem where the initial mesh configuration is invalid, or likewise if at any point in the optimisation procedure this scenario should occur, such as in the reverse line search. To overcome this limitation, we replace J with a regularized version J^r proposed in [13,31], defined as

$$J^r = \frac{1}{2}\left(J + \sqrt{\delta^2 + J^2}\right), \qquad \delta = \begin{cases} \sqrt{\varepsilon_t(\varepsilon_t - \min(J))}, & \text{if } \min(J) < 0 \\ \varepsilon_t, & \text{otherwise} \end{cases}$$

where ε_t denotes a number slightly larger than machine epsilon and $\min(J)$ is the minimum value of the Jacobian in the collection of elements being optimised for a given node. In the case of invalid elements, J^r becomes very small but positive. This alters the profiles of the energy functionals such that they go from being asymptotic to something similar to exponential and thus very large, driving the optimiser to move nodes in the mesh away from small or negative Jacobian regions, correcting the mesh if it were to start to become invalid.

5. Results

In this section, we outline the application of the variational framework to some two- and three-dimensional examples. Throughout this section we consider the generation of isotropic triangular and tetrahedral meshes, which makes the element-wise measure of the scaled Jacobian, defined for an element Ω^e as

$$J_s^e = \frac{\min_{\boldsymbol{\xi}} J(\boldsymbol{\xi})}{\max_{\boldsymbol{\xi}} J(\boldsymbol{\xi})} \quad \forall \Omega_*^e \subset \Omega$$

an appropriate measurement of the element quality. We then define the overall quality of the mesh by considering the minimum scaled Jacobian over the mesh, defined as $J_s = \min_{1 \leq e \leq N_{\mathrm{el}}} J_s^e$. In general an 'ideal' element should be as close to straight-sided as possible. Despite known flaws, such as asymptotic behaviour, it is impartial to all the functionals presented and characterises at least one known criteria of quality in high-order numerical methods: the smoothness of the Jacobian. Therefore results near $J_s^e = 1$ are considered to be the highest quality and any element with $J_s^e < 0$ is an invalid element. All of the tests were performed in parallel using the same 24-core machine, consisting of two Intel(R) Xeon(R) CPU E5-2697v2 processors running at 2.7GHz.

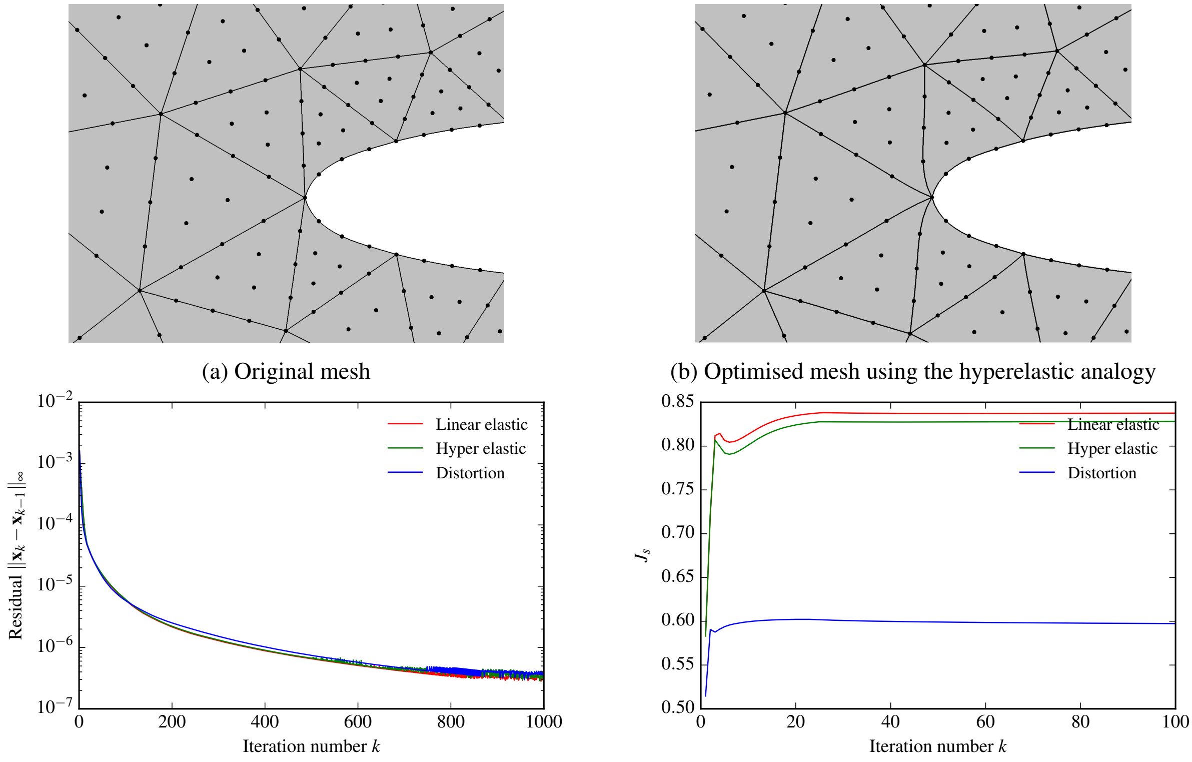
5.1. Parallel efficiency

A very simple example of the node colouring scheme is shown in fig. 3, in which any nodes that have the same colour can be processed in parallel. To demonstrate the efficacy of this very simple colouring, we show the results of a strong scaling simulation that shows the parallel efficiency of the optimisation process when using between 1 and 24 threads. A simple test mesh of a sphere inside a cube was constructed using 10,615 tetrahedra at polynomial order $P = 5$, resulting in 128,254 free nodes, and the time taken to obtain the maximum scaled Jacobian was recorded, occuring within 10 iterations of the optimisation procedure. We observe 76% efficiency between 1 and 24 cores (runtimes 220.5s and 11.9s respectively) and 92% between 1 and 16 cores (runtime 14.9s), which is excellent given the relatively small size of the problem and the simple node colouring strategy used. It is likely that this can be improved further using a more optimal colouring strategy. However for this work the efficiency has substantially reduced the runtime of the optimisation process.

5.2. Two-dimensional examples

To demonstrate the ability of the system to optimise a high-order mesh based on a boundary displacements, we show the results from the smoothing of a 5th order NACA0012 aerofoil mesh comprised of 524 triangles with 6415 free nodes, shown in fig. 4. We see that optimisation from the initial configuration of fig. 4a yields a deformation in elements close to the boundary, as shown in fig. 4b. Since the images only show minor differences, we perform a more quantitative analysis by first visualising the convergence of the residual in fig. 4c. This is deliberately extended to 1,000 iterations, beyond the convergence criterion defined earlier, to examine the long-time convergence properties. The minimum scaled Jacobian across the entire mesh is also shown in fig. 4d to examine the quality of the resulting mesh. The runtime for this problem is 38 seconds. However we should emphasize that it takes fewer than 1,000 iterations to reach an optimal quality,

As noted earlier, the distortion and Winslow methods yield identical results in two dimensions for valid initial configurations. We therefore only show the results from the distortion functional in figs. 4c and 4d. For this simple example, convergence of the residual is broadly identical across all of the considered functionals. However, there is a significant increase in the resulting mesh quality between these different functionals, with the distortion and Winslow functionals yielding a lower quality mesh than the elasticity functionals. This difference in quality is a common theme amongst the results that follow.



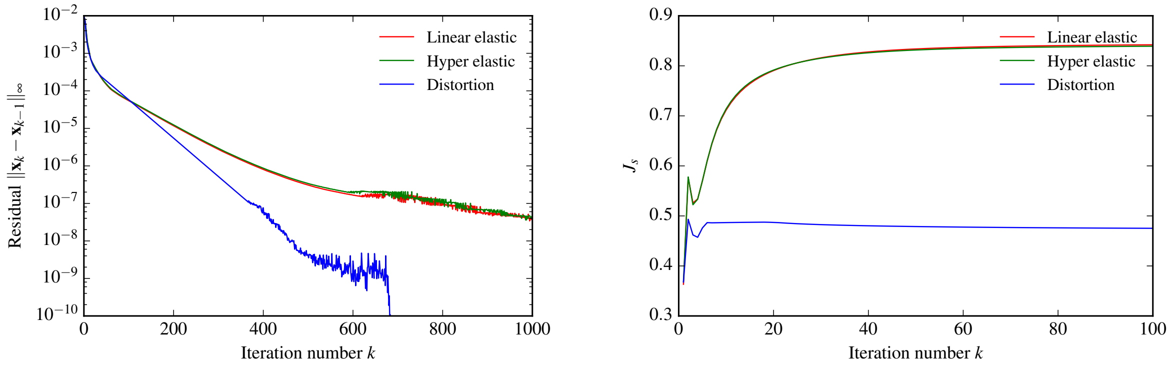
(a) Original mesh

(b) Optimised mesh using the hyperelastic analogy

(c) Convergence of the residual as a function of iteration number

(d) Convergence of the minimum scaled Jacobian

Fig. 4: Application of the variational framework to a triangular mesh of a two-dimensional NACA0012 aerofoil at $P = 5$.

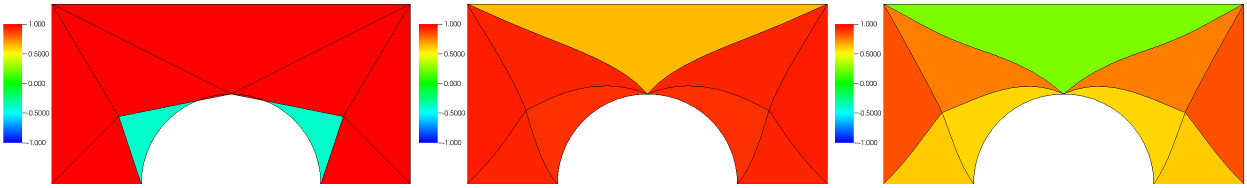


(a) Convergence of the residual as a function of iteration number

(b) Convergence of the minimum scaled Jacobian

Fig. 5: Convergence study of a 10,000 element two-dimensional triangular mesh.

To assess the convergence and performance characteristics of the different energy functionals in more detail, we consider a larger 10,000 element two-dimensional case of a circle of radius 0.25 inside a square of side length 2. Fig. 5a shows the residual of the optimisation process, where the elasticity methods again follow each other closely, reaching a rough convergence point at around 10^{-7}. In the variational setting, the nonlinear and linear elasticity perform at roughly similar levels of computational cost, unlike a traditional Galerkin-based implicit discretisation where one would typically expect the nonlinear case to perform at a slower speed than the linear. However in this case, the distortion method converges far faster than the elasticity methods, taking less than 600 iterations to converge to ap-



(a) Original configuration (b) Optimised with hyperelastic (c) Optimised with distortion

Fig. 6: The optimisation of an initially-invalid two-dimensional example mesh.

proximately 10^{-9}. The convergence curve of the minimum scaled Jacobian in fig. 5b shows that the elasticity methods converge slowly, but to a higher value, which motivates our choice of the stringent residual convergence. However the distortion method reaches a point at which the mesh no longer improves in a matter of only a few iterations. This has interesting implications for computational cost. If we know that after a few iterations the optimisation process will probably have achieved its best result, we need not run it to convergence of the residual, thus drastically reducing overall cost. We observed similar profiles for other 2D cases run during the research for this work, including untangling of the mesh after a handful of iterations.

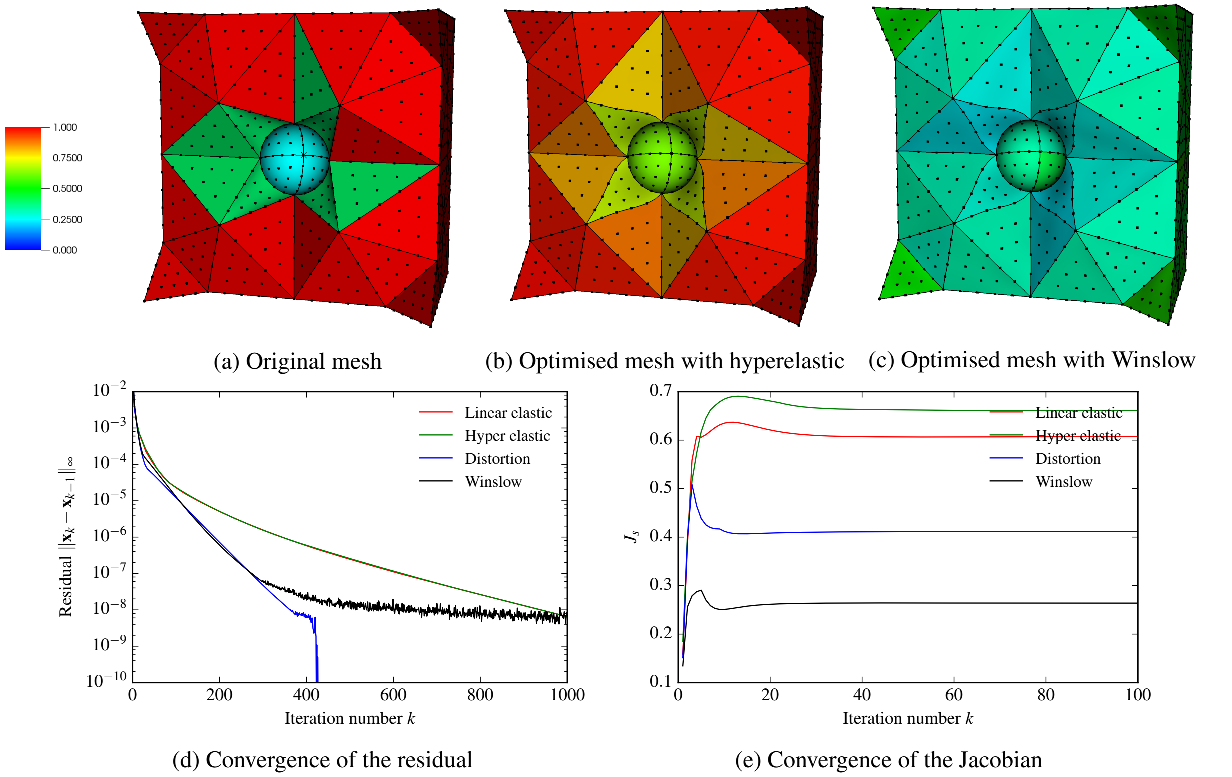
Finally, we illustrate how the variational framework can untangle a initially invalid mesh, shown in fig. 6a, in which the scaled Jacobian of each element is visualised. In this very simple case, an initial mesh of nine triangles, two of which are invalid, are untangled to produce the valid meshes shown in 6b and 6c, showing the capability of the framework to correct invalid elements. Again the elasticity functionals produce a higher quality resulting mesh under these simplified conditions and all successfully untangle the high-order mesh.

5.3. Three-dimensional examples

We now consider the application of the variational framework to three-dimensional geometries. The first example is a canonical sphere-in-cube geometry shown in fig. 7, comprised of 276 tetrahedra at polynomial order $P = 5$, with 4,688 free nodes. We follow a similar methodology as in the previous section, running 1000 iterations of the optimisation procedure to examine the convergence properties of each method and the maximum attainable scaled Jacobian J_s of each functional. Fig. 7 shows the numerical characteristics in 3D for the case presented above. The residual follows a very similar pattern to that observed in the 2D results. The distortion measure converges quickly before exceeding the accuracy of the optimisation method. Once again the elasticity methods are similar. The Winslow method shows similar convergence to the distortion approach before levelling off at a lower value of the residual. In other cases tested during the development of this work, particularly those that have large numbers of elements, this levelling off point can be at a significantly higher value of the residual, meaning that the method exhibits signs of not being able to find a global minimum and is most likely trapped within a local minima. We also visualise the minimum scaled Jacobian of the mesh, which also follows a similar pattern to the two-dimensional examples. This shows that a close-to-optimal mesh is achieved within only a few iterations. Given that the runtime for 1,000 iterations is 124 seconds, this mesh becomes optimised in only a few seconds.

To undertake a wider assessment of the mesh quality distributions, we display the histogram of elemental qualities in fig. 8. In all these histograms the minimum Jacobian is raised from the original distribution, which further demonstrates the ability of the framework to improve mesh quality. We also note that the introduction of curvature into interior elements inevitably leads to a number of elements deviating slightly from the ideal at $J_s = 1$. As can now be expected, we observe a similar pattern between the elasticity approaches and they both achieve similar distributions of element qualities: in this case hyper-elasticity slightly outperforms linear elasticity, but not by a significant amount. The distortion method exhibits a similar distribution to the elasticity methods but the minimum quality is much lower. The Winslow approach improves the minimum quality by a small amount, but leads to the worst overall quality.

As a final demonstration of the capabilities of the framework, we use it to optimise the quality of a large 120,000 tetrahedral element mesh, of a section of the DLR F6 geometry including the engine at polynomial order $P = 4$, as shown in fig. 9. In this figure, we show the surface mesh and the elements connected to the surface that have a scaled



(a) Original mesh (b) Optimised mesh with hyperelastic (c) Optimised mesh with Winslow

(d) Convergence of the residual (e) Convergence of the Jacobian

Fig. 7: Convergence of 3D example of a sphere-in-cube geometry. The colour scale represents values of the scaled Jacobian J_s.

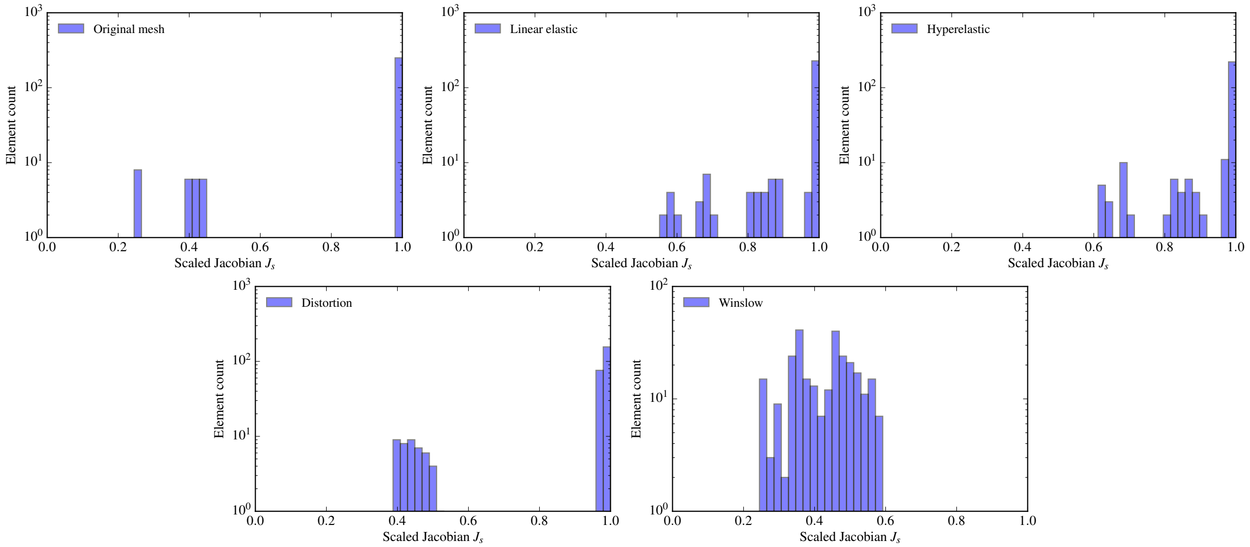


Fig. 8: Histograms of element qualities for the sphere-in-cube geometry. From top left to bottom right: original mesh, linear elastic, hyperelastic, distortion and Winslow functionals.

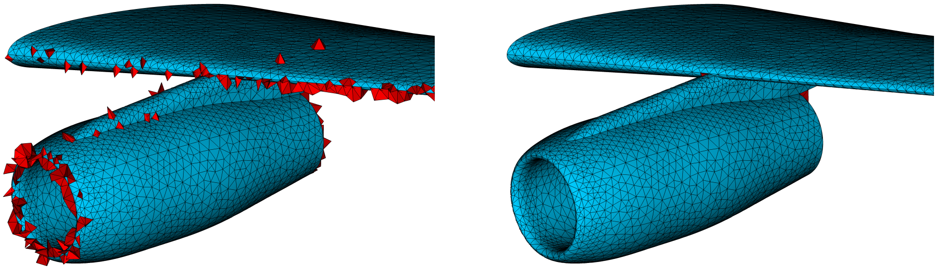


Fig. 9: Optimisation of a DLR F6 wing at polynomial order $P = 4$. On the left, the unoptimised mesh is shown, highlighting those elements that have $J_s^e < 0.5$. On the right, we show the optimised mesh.

Jacobian $J_s^e < 0.5$, both before and after optimisation. In the initial case where only the geometric surface contained curvature, the mesh was invalid with a lowest scaled Jacobian value of -1.02, meaning that the regularisation method was utilised in this case in order to untangle the mesh. Fig. 9 shows the initial configuration, which has 311 elements having $J_s^e < 0.5$, and the same case after optimisation using the hyper-elastic formulation, where this number has been reduced to only 3 elements, with an overall minimum J_s of 0.03. Although this mesh has therefore been successfully made valid, the low J_s can be attributed to low quality linear elements in the rear of the engine where geometric restrictions are severe. Similar behaviour for the other functionals was observed, where the elasticity formulations give roughly equal qualities, followed by the distortion functional and finally the Winslow functional. This case takes around 2 minutes to optimise on our modern workstation.

6. Conclusions

We have presented a parallel variational framework for the optimisation of valid and correction of invalid high-order meshes. We find this to be a capable and efficient approach to the problem at hand and in particular note that a number of the energy functionals used here are known to be polyconvex, convex, or both, meaning that we have theoretical guarantees of a minimum energy in order to improve robustness. In using a variational setting, we are able to reformulate a number of previous methods for curvilinear mesh generation and compare their relative performance. We find that elasticity methods generally produce higher quality meshes – at least under the definition of quality given by the scaled Jacobian – but require more iterations to reach this point of optimisation. In a problem where cost can be prohibitive, use of the distortion or Winslow formulations will converge faster and could therefore be an advantage.

Future work can focus around a number of directions. One disadvantage of the current method is the fixed boundary nodes, which only permits a certain degree of optimisation. Methods that investigate optimisation based on either a sliding node method, as proposed in [32], or through a bottom-up approach whereby this energy method is considered first on curves, followed by surfaces and then volumes, should help to overcome this issue. Additionally, there is scope for further optimisation in considering only regions of the mesh where element quality is low, or to implement this framework in massively parallel GPU architectures or in a distributed cluster setting to further enhance the time-to-compute. Finally, the topic of quality of high-order meshes is highly important and somewhat contentious, since what constitutes ‘high quality’ varies from application to application. We note that this framework is sufficiently flexible to incorporate element quality control by modifying the functional to be optimised. We envisage that this can be done either through the introduction of, for example, thermal stress terms as in previous work [12], or alternatively support the generation of anisotropic meshes where the solution field is highly stretched, through the use of monitor functions, as advocated in [14].

Acknowledgments

MT acknowledges Airbus and EPSRC for funding under an industrial CASE studentship. DM acknolwedges support from the PRISM platform grant under EPSRC award EP/L000407/1.

References

[1] P. Vincent, A. Jameson, Facilitating the adoption of unstructured high-order methods amongst a wider community of fluid dynamicists, Math. Model. Nat. Phenom. 6 (2011) 97–140.

[2] Z. J. Wang, K. Fidkowski, R. Abgrall, F. Bassi, D. Caraeni, A. Cary, H. Deconinck, R. Hartmann, K. Hillewaert, H. T. Huynh, N. Kroll, G. May, P.-O. Persson, B. van Leer, M. Visbal, High-order CFD methods: Current status and perspective, Int. J. Num. Meth. Fluids 72 (2013) 811–845.

[3] S. Dey, R. O'Bara, M. Shephard, Curvilinear mesh generation in 3D, in: Proceedings of the 8th International Meshing Roundtable, 1999. South Lake Tahoe, California.

[4] S. Sherwin, J. Peiró, Mesh generation in curvilinear domains using high-order elements, Int. J. Num. Meth. Eng. 53 (2002) 207–223.

[5] T. Toulorge, C. Geuzaine, J.-F. Remacle, J. Lambrechts, Robust untangling of curvilinear meshes, J. Comp. Phys. 254 (2013) 8–26.

[6] A. Gargallo-Peiró, X. Roca, J. Sarrate, A surface mesh smoothing and untangling method independent of the CAD parameterization, Comp. Mech. 53 (2014) 587–609.

[7] A. Gargallo-Peiró, X. Roca, J. Peraire, J. Sarrate, Defining quality measures for validation and generation of high-order tetrahedral meshes, in: Proceedings of the 22nd International Meshing Roundtable, Springer International Publishing, 2014, pp. 109–126.

[8] X. Roca, A. Gargallo-Peiró, J. Sarrate, Defining quality measures for high-order planar triangles and curved mesh generation, in: Proceedings of the 20th International Meshing Roundtable, Springer Berlin Heidelberg, 2011, pp. 365–383.

[9] Z. Xie, R. Sevilla, O. Hassan, K. Morgan, The generation of arbitrary order curved meshes for 3D finite element analysis, Comp. Mech. 51 (2013) 361–374.

[10] P.-O. Persson, J. Peraire, Curved mesh generation and mesh refinement using Lagrangian solid mechanics, in: 47th AIAA Aerospace Sciences Meeting and Exhibit, Orlando (FL), USA, 2009. AIAA paper 2009–949.

[11] M. Fortunato, P.-O. Persson, High-order unstructured curved mesh generation using the Winslow equations, J. Comp. Phys. 307 (2016) 1–14.

[12] D. Moxey, D. Ekelschot, Ü. Keskin, S. Sherwin, J. Peiró, High-order curvilinear meshing using a thermo-elastic analogy, Comput. Aided Des. 72 (2016) 130–139.

[13] V. Garanzha, Variational principles in grid generation and geometric modelling: theoretical justifications and open problems, Numer. Linear Algebra Appl. 11 (2004) 535–563.

[14] W. Huang, R. Russell, Adaptive moving mesh methods, Springer, 2011.

[15] C. A. Felippa, Optimization of finite element grids by direct energy search, Appl. Math. Model. 1 (1976) 93–96.

[16] S. P. Sastry, V. Zala, R. M. Kirby, Thin-plate-spline curvilinear meshing on a calculus-of-variations framework, Procedia Eng. 124 (2015) 135–147.

[17] L. C. Evans, Partial Differential Equations, volume 19 of *Graduate Studies in Mathematics*, Providence, Rhode Island, 1998.

[18] V. Garanzha, L. Kudryavtseva, S. Utyuzhnikov, Variational method for untangling and optimization of spatial meshes, J. Comput. Appl. Math. 269 (2014) 24 – 41.

[19] J. Bonet, R. Wood., Nonlinear continuum mechanics for finite element analysis, Cambridge University Press, 1997.

[20] R. Hartmann, T. Leicht, Generation of unstructured curvilinear grids and high-order discontinuous Galerkin discretization applied to a 3D high-lift configuration, Int. J. Num. Meth. Fluids (2015).

[21] G. Holzapfel, Nonlinear solid mechanics, Wiley, 2000.

[22] A. Charakhch'yan, S. Ivanenko, A variational form of the Winslow grid generator, J. Comp. Phys. 136 (1997) 385–398.

[23] J. Escobar, E. Rodríguez, R. Montenegro, G. Montero, J. González-Yuste, Simultaneous untangling and smoothing of tetrahedral meshes, Comp. Meth. Appl. Mech. Eng. 192 (2003) 2775–2787.

[24] M. Turner, D. Moxey, S. J. Sherwin, J. Peiró, Automatic generation of 3d unstructured high-order curvilinear meshes, in: Proceedings of the European Congress on Computational Methods in Applied Sciences and Engineering, 2016. Crete Island, Greece.

[25] C. D. Cantwell, D. Moxey, A. Comerford, A. Bolis, G. Rocco, G. Mengaldo, D. de Grazia, S. Yakovlev, J.-E. Lombard, D. Ekelschot, B. Jordi, H. Xu, Y. Mohamied, C. Eskilsson, B. Nelson, P. Vos, C. Biotto, R. M. Kirby, S. J. Sherwin, Nektar++: An open-source spectral/hp element framework, Comp. Phys. Comm. 192 (2015) 205–219.

[26] J. S. Hesthaven, T. Warburton, Nodal Discontinuous Galerkin Methods, Springer, 2008.

[27] A. Heinecke, H. Pabst, G. Henry, LIBXSMM: A High Performance Library for Small Matrix Multiplications, in: Supercomputing 2015, Austin, Texas, 2015.

[28] F. Witherden, P. Vincent, On the identification of symmetric quadrature rules for finite element methods, Comp. Math. Appl. 69 (2015) 1232–1241.

[29] R. Schneider, P. K. Jimack, On the evaluation of finite element sensitivities to nodal coordinates, Electron. Trans. Numer. Anal. 32 (2008) 134–144.

[30] J. Nocedal, S. Wright, Numerical Optimization, Springer series in operations research, second ed., Springer, 2006.

[31] V. Garanzha, I. Kaporin, Regularization of the barrier variational method of grid generation, Comp. Math. Math. Phys. 39 (1999) 1489–1503.

[32] E. Ruiz-Gironés, X. Roca, J. Sarrate, High-order mesh curving by distortion minimization with boundary nodes free to slide on a 3D CAD representation, Comput. Aided Des. 72 (2016) 52–64.